

## Application Guideline - Accessing Databases

**Version:** 1.0

**Adoption Date:** 11/2/2009

**Background Information:** Microsoft has officially released several Object Relational Mapping technologies for database such as LINQ-To-SQL and ADO.NET Entity Framework. These technologies help reduce the amount of code needed to be written in accessing a database via LINQ (Language Integrated Query.) LINQ definitely makes developing applications a more productive process. However, overly using LINQ for complex database functions also creates a dilemma for re-use, maintenance and fine-grained security.

### **Using LINQ to access a database without stored procedures:**

Strategic Architecture Team (SAT) has approved for applications to use LINQ entirely without stored procedures. Developers need to send a request to the ADS Operation Unit for this right. The production connection string for these types of applications will be maintained by the ADS Operations Unit. The password for the production connection string will not be provided to the application developer. If access to production database tables by a developer is deemed necessary another service account could be requested.

### **Using ADO.NET or LINQ for Read access and stored procedures for Create/Update/Delete**

Developers who choose this approach will maintain their own production connection string in their config file. Developers can leverage LINQ by using stored procedures via Linq-to-SQL. Currently at Ecology, this is the preferred method but not mandatory. It provides the benefit of flexibility in LINQ for read actions and leverages the performance and security benefit of stored procedures.

### **Best Practices**

- Use stored procedures if developers are still comfortable with it. A stored procedure still provides the benefit of fast execution, easy to construct and debug, easy to share, and fine-grained security.
- LINQ option can be used for Create/Read/Update/Delete operations without stored procedures where those operations don't require complex join.
- Use stored procedures for complex joins and criteria selections.
- Use stored procedures for reports, BI or heavy data loading/processing.
- For sensitive tables where multi-roles are applied, LINQ should be avoided unless it selects data through views or stored procedures where specific rights are granted to the accounts accessing these views or stored procedures.
- Whether using LINQ or Stored Procedures, developer should not tie the UI tier directly to the data access tier. The UI tier should access data via a middle tier so it can adapt to new data access technology later on and components can be shareable.

## Document History

Date	Version	Editor	Change
10/8/2009	Draft	Son Tran	Replacing LINQ guidance
10/19/2009	Draft		SAT Review
10/22/2009	Draft	Developers and Architecture Workgroup reviewed and Feedback	Changes per feedback
11/2/2009	1.0		SAT Approved